



1-1-2001

A Case Against Software Patents

Russell Moy

Follow this and additional works at: <http://digitalcommons.law.scu.edu/chtlj>



Part of the [Law Commons](#)

Recommended Citation

Russell Moy, *A Case Against Software Patents*, 17 SANTA CLARA HIGH TECH. L.J. 67 (2000).

Available at: <http://digitalcommons.law.scu.edu/chtlj/vol17/iss1/2>

This Article is brought to you for free and open access by the Journals at Santa Clara Law Digital Commons. It has been accepted for inclusion in Santa Clara High Technology Law Journal by an authorized administrator of Santa Clara Law Digital Commons. For more information, please contact sculawlibrarian@gmail.com.

A CASE AGAINST SOFTWARE PATENTS

Russell Moy[†]

TABLE OF CONTENTS

I. Introduction	68
II. Case Study: Growth of Personal Computer Markets Through Software Copyright Protection	70
III. Federal Advisory Commission Recommendations for Intellectual Property Protection of Software	74
A. Presidential Commission on the Patent System.....	74
B. National Commission on New Technological Uses of Copyrighted Works	75
IV. Intellectual Property Statutes: Protection and Promotion of a Software Industry	77
A. Application of the Copyright Statutes to Software.....	77
B. Application of the Patent Statutes to Software.....	80
C. Policy Considerations.....	87
V. Computer Software Distinguished.....	88
VI. Discussion	92
VII. Conclusion	98

[†] B.S.E. (Chemical Engineering) Case Western Reserve University; M.S.E., Ph.D. (Chemical Engineering) The University of Michigan; J.D. Wayne State University School of Law; Licensed Professional Engineer (Michigan). The author is presently a Science and Technology Fellow assigned to the White House Office of Science and Technology Policy within the Executive Office of the President. The views expressed in this article are those of the author and do not necessarily represent the position of the Office of Science and Technology Policy or of any other agency of the U.S. government. The author would like to thank the Society of Automotive Engineers, the Engineering Society of Detroit and Ford Motor Company for the fellowship support he received during the preparation of this manuscript and Jessica Gress for her many valuable editorial comments.

I. INTRODUCTION

Personal computer usage has changed substantially over the past two decades. In 1983 only about seven percent of office workers were using personal computers, and home computers were merely a novelty.¹ Today, the ubiquity of personal computers is apparent in that almost every office and forty-five percent of U.S. homes have a personal computer.² Apple was one of the leading computer manufacturers in the early 1980s.³ Now, it is rare to find a business office that still uses Apple computers. The overwhelming industry leaders are now IBM and IBM compatibles,⁴ such as Compaq, Packard Bell, NEC and Zenith. The growth of these compatibles has more to do with software compatibility rather than hardware superiority and copyright protection of the software was the key to that growth,⁵ whereas, patent protection of software would have frustrated that development.

During this time, intellectual property protection of software also has changed. In the past, algorithms were not considered patentable subject matter.⁶ Over the past two decades however, courts have upheld software and algorithm patents that were integral to the operation of specific hardware devices.⁷ The resulting parallel intellectual property protection for software was inevitable since many software innovations satisfy the statutory definitions for both patents and copyrights. Today, software patents are less dependent on hardware specifics than before since courts now uphold patent protection for generic programs that can operate on a variety of

¹ Robert Blatt, *IBM Predicted to Overtake Apple in Office Mart in '83*, COMPUTERWORLD, Sept. 12, 1983, at 166.

² Alina Matas, *Computer Company Looking for S. Florida Retail Gateways*, BROWARD DAILY BUS. REV., Oct. 7, 1998, at A1, available at LEXIS, News Group File.

³ Andrew Pollack, *Big I.B.M. Has Done it Again*, N.Y. TIMES, Mar. 27, 1983, § 3 (Magazine), at 1. See also Blatt, *supra* note 1.

⁴ INT'L TRADE ADMIN., U.S. DEP'T OF COMMERCE, U.S. INDUS. & TRADE OUTLOOK '98 27-10 (1998).

⁵ See, e.g., John Greenwald, *D-Day for the Home Computer; Marching from Success to Success*, IBM now has a Product for the Living Room, TIME, Nov. 7, 1983, at 76 ("The PC, for example, was not a technological breakthrough and is assembled largely from parts made by outside suppliers.").

⁶ Gottschalk v. Benson, 409 U.S. 63, 65 (1972) ("A procedure for solving a given type of mathematical problem is known as an 'algorithm'.").

⁷ See, e.g., Diamond v. Diehr, 450 U.S. 175 (1981); Arrhythmia Research Technology, Inc. v. Corazonix Corp., 958 F.2d 1053 (Fed. Cir. 1992).

computers. The desire for compatibility and interoperability distinguishes commercial software innovations from what was historically considered a patentable invention. This is because the:

[C]onsumer demand for [software] enjoys positive network effects. A positive network effect is a phenomenon by which the attractiveness of a product increases with the number of people using it. The fact that there is a multitude of people using [a software product] makes the product more attractive to consumers.⁸

The consumer attraction to a particular software product is based therefore on the ability of the consumer to exchange computer application files with the consumer's colleagues.

The Patent Act of 1952 contains a provision that allows for the grant of a patent based on functional claims, which combined with the practical requirement for interoperability, will completely bar a competitor from developing any practical competing products. This is inconsistent with the constitutional intent for the patent system, which is "[t]o promote the Progress of Science and useful Arts."⁹ Consequently, as a matter of public policy, Congress should amend the patent statutes to prohibit the patenting of software and algorithms to be consistent with this constitutional intention, rather than rewarding inventors and patent assignees an absolute monopoly. The patent laws, like those for copyright, grant limited monopolies to the innovators who publicly disclose the details of their innovations to encourage "the social advantages resulting from . . . building on the work of another."¹⁰ However, the virtual nature¹¹ of software innovations, combined with the functional claiming permitted for patents, eliminates any practical opportunity for one to build on the work of another.

Section II illustrates the rising debate between patent and copyright protection for rapid technological advancements through the use of a case study. Section III develops the argument that Congress does not intend that software should be protected by patent.

⁸ United States v. Microsoft Corp., 84 F. Supp. 2d 9, 20 (D.D.C. 1999) (Microsoft Trial Findings of Fact — Fact 39). Here, I substitute "software" and "software product" for "Windows" used in Judge Jackson's findings of fact.

⁹ U.S. CONST. art. I, § 8.

¹⁰ See, e.g., S. Carran Daughtrey, *Reverse Engineering of Software for Interoperability and Analysis*, 47 VAND. L. REV. 145, 146 n.1 (1994).

¹¹ Virtual is defined in the software context as "[n]ot physically existing but made by software to appear to do so from the point of view of the program or the user." 2 THE NEW SHORTER OXFORD ENGLISH DICTIONARY 3586 (1993).

Section IV discusses the statutory schemes that are presently available for protecting software. Section V describes how the nature of software makes it fundamentally different from the other types of creative works. Section VI develops the reasoning for barring software from patent protection. Concluding remarks are presented in Section VII.

II. CASE STUDY: GROWTH OF PERSONAL COMPUTER MARKETS THROUGH SOFTWARE COPYRIGHT PROTECTION

When IBM introduced its first personal computer in August 1981 the market was already cluttered with over 150 competitors.¹² Within two years, IBM owned over twenty-six percent of the market.¹³ The following factors contributed to IBM's success: (1) the use of a more powerful processor than the competition¹⁴; (2) enlisting a wide scale retail distribution network¹⁵; (3) use of high-speed, low cost automated manufacturing facilities¹⁶ and (4) aggressive pricing.¹⁷ Yet, perhaps the most important distinction between IBM and its competitors was the use of *wide-open software*. The wide-open software standard made it possible for software developers to write programs that were compatible with IBM's PC specifications.¹⁸ The publication of these specifications quickly led to a boom in the software development industry for IBM compatibles.¹⁹

IBM's publication of its design specifications and selection was

¹² *Personal Computers: and the Winner is IBM*, BUS. WK., Oct. 3, 1983, at 76.

¹³ *See id.*

¹⁴ Bobbi Bullard, *Comparing the IBM PC and the TI PC*, BYTE, Nov. 1983, at 233.

¹⁵ *See Personal Computers: and the Winner is IBM*, *supra* note 12, at 76, 78.

¹⁶ *See id.*

¹⁷ *See id.*

¹⁸ *See id.* ("IBM published the PC specifications so that people outside the company could write software for the machine. And it allowed Microsoft Corp., the supplier of the PC's basic operating system, to license that software to others. Because of the PC's huge sales potential, most software developers started writing programs for it, making it the de facto industry standard.").

¹⁹ *IBM's Personal Computer Spawns an Industry*, BUS. WK., Aug. 15, 1983, at 88.

IBM laid the cornerstone for [the] PC-compatible market in 1981 . . . by publishing the PC's technical specifications showing how the machine was built and how it operated. This allowed other manufacturers to write applications software and make additional products for it [Although] IBM had always fiercely guarded the designs of its products in traditional mainframe markets[,] . . . it now realized that to sell large numbers of its Personal Computer, it would need an army of programmers outside IBM to make its desktop model dominant. The decision to publish the design was fundamental to [their] success.

Id.

the first important step in "sett[ing] a standard in an unstandardized industry."²⁰ It "brought a focus and a direction to what had seemed a chaotic industry."²¹ By 1983 "[v]irtually every software company [was] giving first priority to writing programs for the I.B.M. machine."²² In March 1982 eighty-five percent of the software sold was for Apple computers and only five percent was for the IBM. Eighteen months later, IBM's twenty-nine percent market share for computer software overtook that for Apple computer software by three percentage points.²³ IBM had become the platform of choice, and more than twenty manufacturers began offering similarly designed clones that were based on the same microprocessor and operating system.²⁴ Unfortunately, some of these clones were only almost interchangeable with the IBM.²⁵ This slight incompatibility meant that some software written for the IBM PC would not run on another PC, despite the common microprocessor and operating system.²⁶ IBM's production capacity in 1983 could not keep up with customer demand. Complete standardization was needed to encourage further hardware and software growth.²⁷

Nevertheless, standardization was not to come from IBM, but from IBM's competitors, despite IBM's efforts to prevent competitors' innovations. Ironically, IBM took legal action against their competitors in twenty-four different suits for patent or copyright infringement of its PC despite the success of its so-called *open architecture policy*.²⁸ In particular, IBM brought actions against several computer companies for plagiarizing IBM's copyrighted ROM BIOS.²⁹ IBM chose to protect the intellectual property of their BIOS code under the copyright laws, undoubtedly because at that time they believed that patent protection for software was unavailable. In this instance, the copyright laws were effective in preventing those

²⁰ See *id.*

²¹ Abby Solomon, *Industries in Transition*, INC., Dec. 1983, at 136.

²² Pollack, *supra* note 3, § 3 at 1.

²³ Greenwald, *supra* note 5, at 77. See also *Personal Computers: and the Winner is IBM*, *supra* note 12, at 76, 78.

²⁴ Pollack, *supra* note 3.

²⁵ *Id.*

²⁶ Bullard, *supra* note 14, at 232.

²⁷ *IBM's Personal Computer Spawns an Industry*, *supra* note 19.

²⁸ Tom McCusker, *Bolt of Lightning: a Decision Concerning Copyright Infringement May Descend on Clone Makers Like the Plague*, *Computer Industry*, DATAMATION, Nov. 1, 1986, at 41.

²⁹ James Langdell, *Phoenix Says its BIOS May Foil IBM's Lawsuits*, PC MAG., July 10, 1984, at 56.

who made no development contribution from profiting from IBM's innovation.

The ROM BIOS is a software program that is "'burned' into chips installed inside each computer, where it controls access to the different functions of the machine"³⁰ by linking a computer's software to its hardware.³¹ "Without a [compatible] BIOS, a clone can't use software written for IBM machines."³² A compatible BIOS needed to be functionally equivalent to the IBM BIOS, yet at the same time, not infringe on IBM's intellectual property rights. Development of the compatible BIOS was essential to the standardization of the personal computer.

Phoenix Technologies announced that it developed an IBM-compatible BIOS in May 1984.³³ Unlike other competitors, Phoenix's programmers were able to satisfy the compatibility and copyright constraints by using so-called *clean room* techniques.³⁴

A 'clean room' is a technique used in the software industry to prevent the direct copying of a competitor's code during the development of a competing product. The procedure usually consists of two teams of developers, one team disassembles the code and describes its functional aspects, while the other team takes the descriptions of the functional aspects and writes the competing product's code. Ideally, this process represents the optimal way to develop a competing product because the alleged infringer can demonstrate that the programmer who drafted the competing code had no access to the original copyrighted work. By showing no access, the alleged infringer could defeat the first requirement of a copyright infringement action and thereby end both the analysis and the case.³⁵

Phoenix's clean room approach consisted of an engineering team in Texas that examined the BIOS software documented in IBM's Technical Reference manual, and wrote a set of specifications that described how the program functioned, without including any actual

³⁰ *Phoenix Technologies Settles ROM BIOS Piracy Suit Against Canadian Manufacturer*, BUS. WIRE, Feb. 23, 1987, available at LEXIS, News Library, News Group File, All File.

³¹ Leslie Helm, *IBM's 'Clone Killers' Don't Scare Phoenix Technologies*, BUS. WK., Dec. 21, 1987, at 113.

³² See *id.* See also Bullard, *supra* note 14, at 236.

³³ Sam Whitmore, *PC-Compatible ROM BIOS Emerges from Phoenix*, PC WK., May 8, 1984, at 5, available at LEXIS, New Group File. See also J. Dickinson, *People in the News: Phoenix Software*, PC MAG., Apr. 30, 1985, at 55, available at LEXIS, News Group File.

³⁴ See, e.g., *The 'Clean Room' Approach*, BYTE, Apr. 1989, at 308.

³⁵ *DSC Communications Corp. v. DGI Techs.*, 898 F. Supp. 1183, 1189 n.3 (N.D. Tex. 1995).

examples of IBM code.³⁶ These specifications were given to a single programmer in Massachusetts who had no experience with the IBM's microprocessor.³⁷ Another Phoenix employee acted as a gatekeeper to route formal questions so as to ensure that the engineers in Texas did not give the programmer in Massachusetts any material that might infringe IBM copyrights.³⁸ "A third group tested the Phoenix software against a variety of programs that ran on IBM's computer."³⁹ Phoenix engineers created an evidentiary audit trail nearly five thousand pages long to document the process in the event that they were challenged in an infringement suit.⁴⁰ Phoenix also took sworn affidavits that its programmer had never seen the source code for IBM BIOS and even offered IBM a chance to examine their code.⁴¹ Phoenix had been so thorough that IBM has never challenged them in court.⁴² Nevertheless, Phoenix had taken out a two million dollar policy from the Hartford Group to protect themselves and their customers against any charge of copyright infringement.⁴³

The availability of the compatible Phoenix BIOS was crucial to the growth of the personal computer industry. Within four years of its introduction, the Phoenix BIOS was more widely used than the IBM BIOS, having been installed in nearly twelve million computers manufactured by over 100 companies.⁴⁴ Phoenix Technologies' growth is perhaps the best example of the success of its BIOS. From 1984, when it introduced its compatible BIOS to 1988, Phoenix grew from a two million dollar to a forty-five million dollar business.

Notwithstanding copyright coverage of software's expressive aspects, to some, the functional quality of software seemed to beg for parallel protection from the patent system. Over the years, two separate federal advisory commissions were convened to provide lawmakers and policy makers with information they could use to balance issues raised in the debate about whether software programs

³⁶ See Langdell, *supra* note 29. See also *How to Copy Software*, THE ECONOMIST, Sept. 16, 1989, at 81.

³⁷ ECONOMIST, *supra* note 36.

³⁸ Steven Burke, *Court Support for 'Clean Room' Cloning May Legalize Intel '386 Chip' Work-Alikes*, PC WK., Feb. 27, 1989, at 63, available at LEXIS, News Group File.

³⁹ ECONOMIST, *supra* note 36, at 81.

⁴⁰ Burke, *supra* note 38, at 63.

⁴¹ Jim Forbes, *Vendors Cook Up Clones; The IBM PC AT Appears to be the Technology to be Followed*, INFOWORLD, Apr. 29, 1985, at 69.

⁴² ECONOMIST, *supra* note 36, at 81.

⁴³ See Langdell, *supra* note 29, at 56; Forbes, *supra* note 43, at 69.

⁴⁴ Ed Scannell, *ROM BIOS Puts Phoenix on Top of Its Market; Provides Link to IBM PC World*, INFOWORLD, Sept. 19, 1988, at 45.

should be patentable.

III. FEDERAL ADVISORY COMMISSION RECOMMENDATIONS FOR INTELLECTUAL PROPERTY PROTECTION OF SOFTWARE

Congress created the U.S. Patent and Copyright Acts pursuant to the constitutional authorization "to promote the Progress of Science and useful Arts."⁴⁵ Since "[t]he Supreme Court treats patents and copyrights the same when looking at the purposes behind the constitutional provision," public policy should protect intellectual property by allocating copyright or patent coverage based considerations that most effectively promote this progress.⁴⁶ "[T]his task involves a difficult balance between the interests of authors and inventors in the control and exploitation of their writings and discoveries on the one hand, and society's competing interest in the free flow of ideas, information, and commerce on the other"⁴⁷

A. Presidential Commission on the Patent System

New technologies raise questions about how these interests should be balanced. In 1965 President Johnson observed that the general character of the patent system had undergone no substantial change since 1836.⁴⁸ Thus, President Johnson convened a Presidential Commission to:

recommend to the President steps to ensure that the patent system will be more effective in serving the public interest in view of the complex and rapidly changing technology [The Commission was instructed to] direct its efforts toward (1) ascertaining the degree to which [the existing] patent system . . . serve[d] . . . national needs and international goals, (2) identify[] any aspects of the system which may [have] need[ed] change, (3) devis[e] possible improvements in the system, and (4) recommend[] any legislation deemed essential to strengthen the United States patent system. In carrying out its evaluation, and in achieving these objectives, the Commission [was required to] make an independent study of the existing patent system of the United States including its relationship to international and foreign patent systems, inventive activity and the administration of the system.⁴⁹

⁴⁵ U.S. CONST. art. I, § 8.

⁴⁶ *Cass County Music Co. v. C.H.L.R., Inc.*, 88 F.3d 635, 641 (8th Cir. 1996).

⁴⁷ *Sony Corp. of Am. v. Universal City Studios, Inc.*, 464 U.S. 417, 429 (1984).

⁴⁸ Exec. Order No. 11215, (Apr. 10, 1965), 1965 WL 7882.

⁴⁹ *See id.*

To summarize, with regard to software, the Commission explicitly recommended against patent protection by stating the following:

A series of instructions which control or condition the operation of a data processing machine, generally referred to as a 'program,' shall not be considered patentable regardless of whether the program is claimed as: (a) an article, (b) a process described in terms of the operations performed by a machine pursuant to a program, or (c) one or more machine configurations established by a program.⁵⁰

This recommendation was based on the Commission's doubt that patents were statutory subject matter, and a belief that it would be infeasible to determine software prior art.⁵¹ The Commission believed patent protection of software was unnecessary because the existing copyright laws offered adequate protection of software. In support of this argument, the Commission noted "that the creation of [software] programs [had] undergone substantial and satisfactory growth in the absence of patent protection and that copyright protection for programs [was] presently available."⁵²

B. National Commission on New Technological Uses of Copyrighted Works

A decade later, Congress convened the National Commission on New Technological Uses of Copyrighted Works (CONTU). The purpose of CONTU is as follows:

to provide the President and the Congress with recommendations concerning those changes in copyright law or procedure needed both to assure public access to copyrighted works used in conjunction with computer and machine duplication systems and to respect the rights of owners of copyright in such works while considering the concerns of the general public and consumer.⁵³

Congress provided interim protection of software through § 117 of the Copyright Act of 1976 pending the Commission's final report.⁵⁴ The legislative history of § 117 highlights congressional

⁵⁰ PRESIDENT'S COMMISSION ON THE PATENT SYSTEM, TO PROMOTE THE PROGRESS OF . . . USEFUL ARTS: IN AN AGE OF EXPLODING TECHNOLOGY at 12. (1966).

⁵¹ *Id.* at 13.

⁵² *See id.*

⁵³ NATIONAL COMMISSION ON NEW TECHNOLOGICAL USES OF COPYRIGHTED WORKS, FINAL REPORT at 1 (1978) [hereinafter *CONTU Report*].

⁵⁴ Pub. L. No. 94-553, 90 Stat. 2565 (1976) (amended by 17 U.S.C. § 117) (Supp. IV 1994).

difficulty in defining a mechanism to protect the rights of software authors.⁵⁵

In its final report, CONTU recommended to Congress that copyright, rather than patent law, should be used to protect software authors and advocated the adoption of a new § 117 within the Copyright Act.⁵⁶ The Commission's reasoning was based in part on its doubt that software could be adequately protected within the Patent Act of 1952.⁵⁷ However, the Commission also believed that the Copyright Act provided appropriate protection for software. The Commission proposed the following guidelines for software protection.⁵⁸

1. Copyright should proscribe the unauthorized copying of these works.
2. Copyright should in no way inhibit the rightful use of these works.
3. Copyright should not block the development and dissemination of these works.
4. Copyright should not grant anyone more economic power than is necessary to achieve the incentive to create.

In this context, the existing § 117 of the Copyright Act must be read in conjunction with § 106, which already protects the software author's monopoly of his creation by prohibiting unauthorized derivative works.⁵⁹ Notwithstanding the recommendations of these

⁵⁵ [I]t has become increasingly apparent that in one major area the problems are not sufficiently developed for a definitive legislative solution. This is the area of computer uses of copyrighted works: the use of a work 'in conjunction with automatic systems capable of storing, processing, retrieving, or transferring information.' The Commission on New Technological Uses is, among other things, now engaged in making a thorough study of the emerging patterns in this field and it will, on the basis of its findings, recommend definitive copyright provisions to deal with the situation.

....

[S]ince it would be premature to change the existing law on computer uses at present, the purpose of section 117 is to preserve the status quo. It is intended neither to cut off any rights that may now exist, nor to create new rights that might be denied under the Act of 1909 or under common law principles currently applicable.

See H.R. REP. NO. 1476, at 116, reprinted in U.S.C.A.N., at 5731 (1976).

⁵⁶ CONTU Report, *supra* note 53, at 29-30.

⁵⁷ *Id.* at 22-23.

⁵⁸ *Id.* at 29.

⁵⁹ 17 U.S.C. § 106 (1994 & Supp. II 1997) (amended by Pub. L. No. 106-44, § 1(g)(2), 113 Stat. 222 (1999)). See also Arthur R. Miller, *Copyright Protection for Computer Programs*,

two commissions and numerous commentaries to the contrary,⁶⁰ it is now accepted that the wording of both the current Patent and Copyright Acts apply to software.

IV. INTELLECTUAL PROPERTY STATUTES: PROTECTION AND PROMOTION OF A SOFTWARE INDUSTRY

Although the subject matter distinction between the U.S. Patent and Copyright systems usually is intuitive, computer software is one area where there continues to be some confusion over which form of protection is most consistent with the intent of the U.S. Constitution.⁶¹ This is because software falls into the statutory definition of both patents and copyrights. Software is a written expression, making it subject to Section 102 of the Copyright Act.⁶² However, it is also subject to Section 101 of the Patent Act⁶³ because it is useful. As a result, patent and copyright protection both have been assigned to computer software, with the application of the Patent Act for computer software evolving over the past three decades.⁶⁴

A. Application of the Copyright Statutes to Software

Copyright and patent laws protect different aspects of intellectual property. "The purpose of copyright is to grant authors a limited property right in the form of expression of their ideas."⁶⁵ The strength of copyright to protect software comes from "[t]he text of the current Copyright Act [which] rejects any general freedom to decompile."⁶⁶

Databases and Computer Generated Works: Is anything new Since CONTU?, 106 HARV. L. REV. 977 (1993).

⁶⁰ See, e.g., Miller, *supra* note 59, at 977; Pamela Samuelson, *Benson Revisited: The Case Against Patent Protection for Algorithms and Other Computer-Related Inventions*, 39 EMORY L.J. 1025 (1990).

⁶¹ Cf. U.S. CONST. art. I, § 8. The U.S. patent and copyright laws are rooted in the U.S. Constitution, "Congress shall have the power . . . To promote the progress of science and useful arts, but securing for limited times to authors and inventors the exclusive right to their respective writings and discoveries." See also Miller, *supra* note 59, at 977; see also Samuelson, *supra* note 60, at 1025; see generally *CONTU Report*, *supra* note 53.

⁶² 17 U.S.C. § 102 (1994) (excluding copyright protection "to any idea, procedure, process, system, method of operation, concept, principle or discovery.").

⁶³ 35 U.S.C. § 101 (1994).

⁶⁴ See generally *Gottschalk v. Benson*, 409 U.S. 63 (1972); *Parker v. Flook*, 437 U.S. 584 (1978); *Diamond v. Diehr*, 450 U.S. 175 (1981); *In re Alappat*, 33 F.3d 1526 (Fed. Cir. 1994); *In re Lowry*, 32 F.3d 1579 (Fed. Cir. 1994); *State St. Bank & Trust Co. v. Signature Fin. Group*, 149 F.3d 1368 (Fed. Cir. 1998); *AT&T Corp. v. Excel Communications, Inc.*, 172 F.3d 1352 (Fed. Cir. 1999).

⁶⁵ *CONTU Report*, *supra* note 53, at 32.

⁶⁶ Miller, *supra* note 59, at 1014. Professor Miller was appointed to the Software Subcommittee

"Decompilation is the process of analyzing a machine-language computer program (a format comparatively few people can read and comprehend) and re-rendering it in a human-readable form."⁶⁷ This human-readable form could be used to facilitate the development of a competing product with only a limited investment, which could be sold "as a less expensive commercial substitute for the original."⁶⁸

[D]ecompile allows a second comer to create a market substitute and reap the benefits of a successful program after others have incurred the risk and expense of its development—an especially inappropriate result given the extraordinary discrepancy between the cost of creating the software and the cost of duplicating it.⁶⁹

"The limited case law in this area seems to establish a presumption that decompilation is itself an infringement, independent of any finding of substantial similarity between the defendant's final product and the original."⁷⁰ The prohibition against any decompilation of computer programs on the face of the Copyright Act is an important aspect of copyright protection for software "because the traces of copying can be disguised, [by an infringer who can] electronically massage the copy until every trace of that illicit reproduction is obscured."⁷¹

But the copyright statute raises another barrier to the protection of software. On the face of the copyright statute there is an apparent conflict between the simultaneous patentability and copyrightability of software. Section 102(b) of the Copyright Act⁷² specifically precludes copyright protection of "any idea, procedure, process, system, method of operation, concept, principle or discovery." This falls within the statutory definition of patentable subject matter. As such, some commentators suggest that computer software might be

of the National Commission on New Technological Uses of Copyrighted Works by President Ford, "which had primary responsibility for studying the relationship between software and copyright and for preparing the portion of the CONTU Final Report that addressed computer programs." *Id.* at 981. See also 17 U.S.C. § 106(2) (1994 & Supp. II 1997) (*amended by* Pub. L. No. 106-44, § 1(g)(2), 113 Stat. 222 (1999)) ("[T]he owner of copyright under this title has the exclusive rights to do and to authorize any of the following . . . to prepare derivative works based upon the copyrighted work.").

⁶⁷ Miller, *supra* note 59, at 1014.

⁶⁸ See *id.*

⁶⁹ *Id.* at 1026.

⁷⁰ *Id.* at 1014.

⁷¹ *Id.* at 1026-1027.

⁷² 17 U.S.C. § 102(b) (1994).

barred from copyright coverage.⁷³ Despite this apparent inconsistency, the Supreme Court has held that a given work can be protected simultaneously by patent and copyright.⁷⁴ Congress has afforded copyright protection to items that are both functional and expressive as early as 1790.⁷⁵ “[N]othing in the copyright statute support[s] the argument that the intended use or use in industry of an article eligible for copyright bars or invalidates its registration [as a patent].”⁷⁶

It is now well-established under the amended 1976 [Copyright] Act that a computer program is a ‘work of authorship’ and is subject to copyright protection. Under the Act, computer programs are classified as ‘literary works.’ Case law under the Act also clearly establishes that copyright protection extends to both a program’s source code, written in conventional human language and symbols, and object code, written in machine readable binary language.⁷⁷

Note that copyright confers protection only for the statements of the computer program and not its function.

The copyright of a work . . . cannot give the author an exclusive right to the methods of operation which he propounds . . . so as to prevent an engineer from using them whenever occasion requires . . . [T]he teachings of science and the rules or methods of useful art [patents] have their final end in application and use; and this application and use are what the public derive from the publication But as embodied in and taught in a literary composition or book, their essence consists only in their statement. This alone is what is secured by the copyright. The use by another of the same methods or statement, whether in words or illustrations . . . would undoubtedly be an infringement of the copyright [If the concept] was not patented . . . [the concept] . . . is open and free to the use of the public.⁷⁸

Because of its prohibition to decompile, copyright law protects computer programs against wholesale duplication by competitors. Absent additional patent coverage, the functional aspects of a copyrighted computer program can only be recreated legally as

⁷³ See, e.g., Miller, *supra* note 59, at 987.

⁷⁴ Baker v. Selden, 101 U.S. 99, 104-105 (1880).

⁷⁵ See Mazer v. Stein, 347 U.S. 201 (1954).

⁷⁶ *Id.* at 218.

⁷⁷ Digital Communications Assoc., Inc. v. Softklone Distrib. Corp., 659 F. Supp. 449, 454 (N.D. Ga. 1987) (citations omitted).

⁷⁸ Baker, 101 U.S. at 103-104.

Phoenix had done using clean room techniques. One legal commentator who has been involved with such clean room techniques observed:

[I]t would be easier and far less expensive to develop entirely new software, were it not for the need in most such cases to have a functional equivalent, compatible program that cannot be obtained in any other way. This is especially so in cases of a need for 'interoperability,' such as running two independently developed programs, or two hardware systems from different suppliers, together.⁷⁹

In the case of Phoenix, half as much would have been spent on cloning the IBM BIOS had a clean room not been necessary.⁸⁰ That clean room cost undoubtedly discourages many competitors from developing functionally equivalent software. Consequently, copyright protection has proven to be effective in protecting software by promoting both the arts and sciences and simultaneously protecting the inventor from competitors by creating sufficient barriers to entry and thus eliminating the need for patent protection in addition to copyright protection.

B. Application of the Patent Statutes to Software

Patents protect different intellectual property aspects of a creation than copyrights. "Patents are designed to give inventors a short-term, powerful monopoly in devices, processes, compositions of matter and designs which embody their ideas A [patented] work must be useful, novel and non-obvious to those familiar with the state of the art in which the patent is sought."⁸¹ Patentable subject matter consists of "any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof."⁸² Software programs consist of collections of mathematical relationships. Thus, the application of the definition of patentable subject matter to early software patent applications caused a great deal of confusion.⁸³ Primarily, this was due to early Court decisions' exclusion of "a principle in the abstract" from patent coverage and the

⁷⁹ G. Gervaise Davis III, PLI, SCOPE OF PROTECTION OF COMPUTER-BASED WORKS: REVERSE ENGINEERING, CLEAN ROOMS AND DECOMPIATION, 370 PLI/PAT 115, 151 (1993).

⁸⁰ Burke, *supra* note 38, at 63.

⁸¹ *CONTU Report*, *supra* note 53, at 32, 33.

⁸² 35 U.S.C. § 101 (1994).

⁸³ See, e.g., *Parker v. Flook*, 437 U.S. 584, 595 (1978); *Gottschalk v. Benson*, 409 U.S. 63, 72-73 (1972).

Court refused to grant any exclusive rights to these “fundamental truth[s].”⁸⁴ It is now accepted that software is statutory subject matter as either a machine or process within the present patent statutes. This acceptance was developed through Supreme Court and Federal Circuit case law over the past twenty-seven years.⁸⁵

Patent coverage of software originates in Section 101 of the Patent Act, which provides patent protection for “any new and useful process, machine, manufacture, or composition of matter.”⁸⁶ Nothing in the code or its legislative history explicitly excludes computer programs or algorithms as patentable subject matter, even absent any physical effects or changes. Software has been patented as both processes and machines.⁸⁷ The scope of the patent coverage has broadened with the availability of functional claims because “such claims will be read to cover every conceivable way of achieving that function.”⁸⁸

*Gottshalk v. Benson*⁸⁹ was the Supreme Court’s first attempt to define the patent system’s role in the protection of computer software. In *Gottshalk*, a patent was:

sought . . . on a method [or process] of programming a general-purpose digital computer to convert signals from binary-coded decimal form into pure binary form The procedures set forth the . . . claims [in the form of an algorithm, which is] a generalized formulation for programs to solve mathematical problems of converting one form of numerical representation to another. [It is from this] generic formulation . . . that programs may be developed as specific applications.⁹⁰

Although processes are a patentable subject matter for patents

⁸⁴ *Leroy v. Tatham*, 55 U.S. 155, 175 (1852).

⁸⁵ See *Diamond v. Diehr*, 450 U.S. 175 (1981); *Flook*, 437 U.S. at 595; *Gottschalk*, 409 U.S. at 72-73; *AT&T Corp. v. Excel Communications, Inc.*, 172 F.3d 1352 (Fed. Cir. 1999); *State St. Bank & Trust Co. v. Signature Fin. Group*, 149 F.3d 1368 (Fed.Cir. 1998); *In re Lowry*, 32 F.3d 1579 (Fed.Cir. 1994); *In re Alappat*, 33 F.3d 1526 (Fed. Cir. 1994).

⁸⁶ 35 U.S.C. § 101 (1994).

⁸⁷ See, e.g., *Excel Communications, Inc.*, 172 F.3d 1352; *Signature Fin. Group*, 149 F.3d 1368.

⁸⁸ MARTIN J. ADELMAN, ET AL., *CASES AND MATERIALS ON PATENT LAW* 662 (1998); Section 112, ¶ 6 provides:

An element in a claim for a combination may be expressed as a means or step for performing a specified function without the recital of structure, material, or acts in support thereof, and such claim shall be construed to cover the corresponding structure, material, or acts described in the specification and equivalents thereof.

35 U.S.C. § 112 (1994).

⁸⁹ *Gottschalk*, 409 U.S. at 63.

⁹⁰ *Id.* at 65.

under the Act,⁹¹ the Court held that the mathematical formula claimed was too “abstract and sweeping” for patent coverage.⁹² In reaching this conclusion, the Court noted that since the claimed procedure could be performed even without a computer, it was only an abstract idea, and abstract ideas are not patentable.⁹³ The Court reasoned that “[t]he mathematical formula [claimed had] no substantial practical application except in connection with a digital computer, which meant that if [the patent were allowed, it] would [have] wholly pre-empt[ed] the mathematical formula and in practical effect would be a patent on the algorithm itself.”⁹⁴

The Supreme Court set a high standard of utility for mathematical formulas.⁹⁵ In *Parker v. Flook*, the Court declined to allow patent protection for “the identification of post-solution applications of [a mathematical] formula” even though they found that formula to be “useful.”⁹⁶ Flook had attempted to patent a process for an improved method of using a mathematical formula in the real-time calculation of alarm limits used in a chemical production. In its analysis, the Court assumed *arguendo* that the mathematical “formula [was] novel and useful and that [Flook] had discovered it.”⁹⁷ However, the Court found that “the only difference between the conventional methods of changing alarm limits and that described in respondent’s application rest[ed] in the second step—the mathematical algorithm or formula.”⁹⁸ Thus, the *Flook* Court declined to uphold the patent application because the “the process itself, not merely the mathematical algorithm, must be new and useful.”⁹⁹

Here, the patent application failed to explain how the algorithm could be used “to select the appropriate margin of safety, the weighting factor or any of the other variables.”¹⁰⁰ The Court held that patentability was only based on the novelty of chemical aspects of the process and not the process control when it found the patent did not

⁹¹ 35 U.S.C. § 101 (1994).

⁹² *Gottschalk*, 409 U.S. at 68.

⁹³ *Id.* at 67.

⁹⁴ *Id.* at 71-72.

⁹⁵ 35 U.S.C. § 101 (1994) (“Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent there for, subject to the conditions and requirements of this title”).

⁹⁶ *Parker v. Flook*, 437 U.S. 585, 585 (1978).

⁹⁷ *Id.* at 588.

⁹⁸ *Id.* at 585-86.

⁹⁹ *Id.* at 591.

¹⁰⁰ *Id.* at 586.

“contain any disclosure relating to the chemical processes at work, the monitoring of process variables, or the means of setting off an alarm or adjusting an alarm system.”¹⁰¹ The Court found the utility of the claimed mathematical formula was limited to the control of the process because it only described “a formula for computing an updated alarm limit, [which could have also been] made by pencil and paper calculations.”¹⁰² The Court did not accept Flook’s argument that his inclusion of specific post-solution activity could be distinguished from the *Benson* case. The *Flook* Court held that “[t]he notion that post-solution activity, no matter how conventional or obvious in itself, can transform an unpatentable principle into a patentable process exalts form over substance.”¹⁰³

In 1981 the Court seemingly overruled these cases *sub silentio* by finding that the claims in a patent application by Diehr and Lutton (*Diamond v. Diehr*) did provide the necessary nexus between an algorithm and a chemical process.¹⁰⁴ According to the Court, this combination satisfied the statutory subject matter requirement.¹⁰⁵ Here, the claimed invention related to an improved method for curing rubber. Although the chemistry of the processes for curing rubber was well-known, acceptable control of those processes was highly dependent on the particular mold geometry, activation energy for curing particular materials and actual temperature of the mold.¹⁰⁶ While mold geometry and activation energy are constant for a given process, the mold temperature varies with time.¹⁰⁷

The relationship between chemical reaction time and temperature is based on the well-known Arrhenius equation, which can be used to calculate appropriate reaction times for a constant mold temperature.¹⁰⁸ As a practical matter, one is precluded from making those calculations manually because of the continuous variation of mold temperature. The patent application litigated by Diehr and Lutton involved the means to determine the mold temperature using a computer to make frequent real-time Arrhenius calculations to

¹⁰¹ *Id.*

¹⁰² *Flook*, 437 U.S. at 585.

¹⁰³ *Id.* at 590.

¹⁰⁴ *Diamond v. Diehr*, 450 U.S. 175 (1981). A patent was subsequently issued. See U.S. Patent No. 4,344,142 (issued Aug. 10, 1982).

¹⁰⁵ See *id.*

¹⁰⁶ *Id.* at 177.

¹⁰⁷ *Id.* at 178.

¹⁰⁸ *Id.* at 177 n.2.

determine status of the reaction.¹⁰⁹ A computer sent a signal to open the mold when the curing reaction was complete. Thus, the use of the computer significantly reduced the chances for overcuring or undercuring.

Although the curing of rubber was previously known and the Arrhenius equation alone did not constitute patentable subject matter, the Court held that the combination taught in *Diehr* was patentable.¹¹⁰ Process patent claims may consist of a "new combination of steps in a process . . . even though the discrete constituent parts of the patent might be well known and in common use before the combination was made."¹¹¹ A process "claim containing a mathematical formula" will satisfy the requirements of Section 101 if it "implements or applies that formula in a structure or process which, when considered as a whole, is performing a function which the patent laws were designed to protect."¹¹² In this case, the claims of the patent were not intended to patent a mathematical formula, rather, as the Court emphasized, the patent's claims sought to protect an industrial process for the molding of rubber products. Thus, the process met the requirements set forth in Section 101 of the statute and was therefore patentable.¹¹³

In the *Benson*, *Flook*, *Diehr* trilogy, the Supreme Court determined that software is patentable if it is part of a structure of a physical process, when considered as a whole. Software or algorithms unconnected to a patentable physical process are not patentable. Moreover, the mere recitation of post-specific activity will not transform an otherwise unpatentable program into one that is patentable.

These Supreme Court holdings seem to indicate caution about granting patent protection to software. In contrast, the Court of Appeals for the Federal Circuit has not been shy about granting patent protection to software-related patent applications. In *In re Alappat*, the appellate court overturned the Board of Patent Appeals and Interferences' denial of a patent for an improved display on an oscilloscope.¹¹⁴

An oscilloscope is a device used to display a continuum of values

¹⁰⁹ *Id.* at 180 n.5.

¹¹⁰ *Diehr*, 450 U.S. at 183 n.8.

¹¹¹ *Id.* at 188. "The 'novelty' of any element or steps in a process, or even of the process itself, is of no relevance in determining whether the subject matter of a claim falls within § 101 categories of possibly patentable subject matter." *See id.* at 188-89.

¹¹² *Id.* at 192.

¹¹³ *Id.* at 192-93.

¹¹⁴ *In re Alappat*, 33 F.3d 1526 (Fed.Cir. 1994).

in a graphical form. Unfortunately, oscilloscopes can only approximate these values because the oscilloscope uses discrete pixels for the display and the result may be discontinuous or jagged. Alappat and his co-workers recognized that these discontinuities could be made less apparent by adjusting the brightness of each pixel based on the quality of the respective approximation. Their invention consisted of an algorithm that reduced the apparent display discontinuity by adjustment of the brightness of the display pixels on an oscilloscope.

The Federal Circuit held in *Alappat* that the Patent and Trademark Office (PTO) must consider 35 U.S.C. § 112, ¶ 6 when evaluating statutory subject matter according to 35 U.S.C. § 101, and found that the patent application described a rasterizer, which is a machine within the statutory definition.¹¹⁵ Furthermore, the court stated that the patent was not barred merely because it could be implemented on a general purpose computer. The court reasoned that “programming creates a new machine, [as] a general purpose computer in effect becomes a special purpose computer once it is programmed to perform particular functions pursuant to instructions from program software.”¹¹⁶ Of course, a similar result could be obtained with a finer pixel resolution oscilloscope. But the patent here describes a method of operating the hardware in a manner to accomplish the same result.

Later that summer, the Federal Circuit decided *In re Lowry*.¹¹⁷ At issue in *Lowry* was whether a new, more efficient format for storing information in a computer memory was barred from patent protection because of obviousness.¹¹⁸ In deciding that it was not, the Federal Circuit first established that computer memory formats were a statutory subject matter by distinguishing computer data formats from printed matter, which is not patentable. The court then held the format was not obvious. Though the court conceded that “the stored data adopted no ‘physical structure,’ per se,” it rationalized the existence of structure.¹¹⁹ “More than mere abstraction, the data structures are specific electrical or magnetic structural elements in a memory . . . [which] are physical entities that provide increased

¹¹⁵ *Id.* at 1540.

¹¹⁶ *Id.* at 1545.

¹¹⁷ *In re Lowry*, 32 F.3d 1579 (Fed.Cir. 1994).

¹¹⁸ *Id.* at 1582. The utility of the claimed invention was not in dispute. *Id.* at 1580.

¹¹⁹ *Id.* at 1583.

efficiency in computer operation.”¹²⁰

Software patent coverage saw its greatest expansion when the Court of Appeals for the Federal Circuit decided *State Street Bank v. Signature Financial Group, Inc.* In *State Street Bank*, the Federal Circuit held “that the transformation of data by a machine through a series of mathematical calculations into a final [mutual fund] share price, constitutes a practical application of a mathematical algorithm, formula, or calculation, because it produces ‘a useful concrete and tangible result.’”¹²¹ The patent at issue, assigned to Signature Financial Group, involved a method to manage a collection of mutual funds so as to retain autonomy of the individual funds, while providing tax advantages and scale economies of the collection of funds.¹²² The patented method takes into account income, expenses, gains and losses for each of the individual funds and calculates the portion of the portfolio held by each member. This is necessary because each member trades shares on a daily basis and each member’s share value is based on its percentage holding. These calculations must be performed quickly and accurately and a computer is necessary because of their complexity.

In upholding the patent, the Federal Circuit stated that statutory subject matter should not be determined by focusing on the four categories of subject matter recited in 35 U.S.C. § 101, “but rather the essential characteristics of the subject matter, in particular, its practical utility.”¹²³ Here, the court found the invention to describe a *machine*, but indicated that the same conclusion would have been reached if it had been considered a *process*. *State Street Bank* represented a shift in the understanding of what constitutes patentable subject matter. This machine produces merely a number that may consist of price, profit, percentage, cost or loss.¹²⁴ The Signature patent “contain[s] six ‘machine’ claims, which incorporated [only] means-plus-function clauses.”¹²⁵ The Federal Circuit invoked 35 U.S.C. § 112, ¶ 6 to uphold the Signature Financial Group patent, which provides for means-plus-function or step-plus-function claims.¹²⁶ Likewise, the Federal Circuit applied similar reasoning

¹²⁰ *Id.* at 1583-84.

¹²¹ *State St. Bank & Trust Co. v. Signature Fin. Group*, 149 F.3d 1368, 1373 (Fed. Cir. 1998).

¹²² U.S. Patent No. 5,193,056 (issued Mar. 9, 1993).

¹²³ *State St. Bank*, 149 F.3d at 1375.

¹²⁴ *See id.*

¹²⁵ *Id.* at 1371. *See also* U.S. Patent No. 5,193,056 (issued Mar. 9, 1993).

¹²⁶ *State St. Bank*, 149 F.3d at 1371-1372; 35 U.S.C. § 112 ¶ 6 (1994); *see also* 64 Fed. Reg. 41,392, 41,393 (1999).

when it upheld process claims in *AT&T v. Excel Communications, Inc.*¹²⁷

The rulings threw open the gates to [software patent] applications. Submissions for software patents have more than doubled to 2,600 in the past fiscal year, and the number of patents granted has mushroomed from 108 in 1996 to 500 in the 1999 fiscal year, according to the Patent and Trademark Office.¹²⁸

Even though the gates have been thrown open due to these circuit court decisions, it is only Congress that has the power to amend the statutory protection of patents and copyrights as called for by the Supreme Court and federal commissions.

The Patent Act contains a requirement for an examination of an innovation that tests for novelty, utility and obviousness that is not included in copyright registration.¹²⁹ In principle, this examination provides a quality certification of the intellectual property protected by a patent that is unavailable for materials protected under copyright. Unfortunately, the virtual nature of software is incompatible with such an examination procedure,¹³⁰ which compromises the quality certification benefit of the Patent Act. Nevertheless, patents for software based on functionality still grant an absolute monopoly for those functions claimed.¹³¹ The grant of this monopoly is against public policy because it discourages innovation.

C. Policy Considerations

The state of the law on patent protection of software through *Lowry* is summarized in the Patent Office's "Examination Guidelines for Computer-Related Inventions."¹³² It should be noted that Supreme Court dicta twice raised policy questions as to the patentability of computer programs.¹³³ In *Benson*, the Court observed that information beyond that which was litigated was needed to answer the broader question as to whether programs are to be patentable.¹³⁴

¹²⁷ *AT&T Corp. v. Excel Communications, Inc.*, 172 F.3d 1352, 1361 (Fed.Cir. 1999).

¹²⁸ Rajiv Ghandrasekaran, *Y2K Patent Pending Fortune; Holding Rights to a Common Fix, Bruce Dickens is Sending the Bill*, THE WASHINGTON POST, Dec. 17, 1999, at E1.

¹²⁹ 17 U.S.C. 407 (1994 & Supp. III 1997); 17 U.S.C. § 408 (1994); 35 U.S.C. § 101 (1994); 35 U.S.C. § 102 (1994); 35 U.S.C. § 103 (1994 & Supp. III 1997); 35 U.S.C. § 131 (1994).

¹³⁰ See *infra* section VI.

¹³¹ See *infra* text accompanying notes 152-193.

¹³² Examination Guidelines for Computer-Related Inventions, 61 Fed. Reg. 7478 (Feb. 28, 1996).

¹³³ *Parker v. Flook*, 437 U.S. 584, 595 (1978); *Gottschalk v. Benson*, 409 U.S. 63, 72-73 (1972).

¹³⁴ *Gottschalk*, 409 U.S. at 72-73.

The *Flook* Court assumed that the algorithm described in the patent application was novel, useful and attributable to the applicant.¹³⁵ However, the Court used its holding in *Benson* to find that the applicant *discovered*, rather than *invented* the algorithm, which lead to the Court's denial of the application.¹³⁶ While conceding that this reasoning was based on *stare decisis* that predates the conception of "the modern business of developing programs for computers"¹³⁷ the Court observed that the grant of a patent for software would "extend patent rights into areas wholly unforeseen by Congress."¹³⁸ The Court believed that a determination of whether such a grant would be appropriate would require further direction from Congress."¹³⁹

Congress has yet to respond to the recommendations of two federal advisory commissions and the Supreme Court *dicta*. Such a response from Congress would involve an amendment to the Patent Act that would explicitly exclude software from coverage. This change would allow competitors to develop the new, innovative software products that are presently prohibited under the Act. New competitive products give customers more choices at a fair price. Under this change, software developers would still have all of the intellectual property protections offered by the Copyright Act.

V. COMPUTER SOFTWARE DISTINGUISHED

A robust intellectual property system is important to promote innovation and in this regard, computer software is no different. Today, new software does differ from other categories of innovation because it falls under both the Patent and Copyright Acts. Another

It may be that the patent laws should be extended to cover these programs, a policy matter to which we are not competent to speak. . . . If these programs are to be patentable, considerable problems are raised which only committees of Congress can manage, for broad powers of investigation are needed, including hearings which canvass the wide variety of views which those operating in this field entertain. The technological problems tendered in the many briefs before us indicate to us that considerable action by the Congress is needed.

Id.

¹³⁵ *Flook*, 437 U.S. at 588, 592 (construing the "principle or mathematical formula," (the algorithm) to be "well known," based on the prior holding in *Benson*).

¹³⁶ *Id.* at 588.

¹³⁷ *Id.* at 595.

¹³⁸ *Id.* at 596.

¹³⁹ *Id.* at 595 ("Difficult questions of policy concerning the kinds of programs that may be appropriate for patent protection and the form and duration of such protection can be answered by Congress on the basis of current empirical data not equally available to this tribunal.").

distinction is the consumer demand for forward, backward and cross-platform compatibility.¹⁴⁰ This compatibility creates an “intractable ‘chicken-and-egg’ problem” for the introduction of new computer systems or software for a new computer system.¹⁴¹

Such a chicken-and-egg problem exists because any incompatibility might dissuade buyers of software and hardware from purchasing incompatible products.¹⁴² This market limitation tends to

¹⁴⁰ See, e.g., *Zenith Radio to Sell VHS Video Recorder And Drop Beta Line Made by Sony Corp.*, WALL ST. J., Jan. 2, 1984, available at 1984 WESTLAW, Allnewsplus File. The commercial demise of the Sony Beta videotape format was undoubtedly due to its limited compatibility. Other examples include videotapes, DVDs and music storage media like cassette tapes, CDs, 8-track tapes and phonograph records. Reverse compatibility has generally not been maintained in these ‘entertainment’ storage media. For example, CD players cannot be used with phonograph records, and DVD players cannot be used with videotapes. With regard to e-commerce technologies, Raymond Kammer, Director of the National Institute of Standards and Technology testified “[t]he success of these new technologies is closely linked to the development of interoperability standards, which specify how devices communicate with each other. Successful standardization efforts in this area are global and driven by technical superiority.” See also *The Role of Standards in Today’s Society and in the Future, 2000: Hearings Before Subcomm. on Technology of the House Comm. on Science*, 106th Cong. (2000) (statement of Raymond G. Kammer, Director, National Institute of Standards and Technology), available at http://www.house.gov/science/kammer_091300.htm [hereinafter *Kammer*]. (noting further that U.S. “standards are developed through a . . . system administered by the private sector . . . [to meet] domestic marketplace needs on a sector-by sector basis.”).

¹⁴¹ *United States v. Microsoft Corp.*, 84 F. Supp. 2d 9, 18 (D.D.C. 1999). In Finding of Fact 35, Judge Jackson establishes that the Windows operating system is the “dominant, persistent and increasing” PC operating system. *Id.* at 19. (In this portion of the discussion, I substitute ‘PC’ and ‘system’ for ‘Intel-compatible PC’ and ‘Intel-compatible PC operating system’ used in Judge Jackson’s findings of fact.)

¹⁴² See *id.* In the case of computer hardware,

[t]he overwhelming majority of consumers will only use a PC . . . for which there already exists a large and varied set of high-quality, full-featured applications, and for which it seems relatively certain that new types of applications and new versions of existing applications will continue to be marketed at pace with those written for other [PCs]. . . . Users do not want to invest in [a PC] until it is clear that the system will support generations of applications that will meet their needs, and developers do not want to invest in writing . . . applications . . . for a [PC] until it is clear that there will be a sizable and stable market for it. What is more, consumers who already use one [compatible PC] are even less likely than first-time buyers to choose a newcomer to the field, for switching to a new system would require these buyers to scrap the investment they have made in applications, training and certain hardware.

....

[C]onsumer interest in a PC . . . derives primarily from the ability of that system to run applications. The consumer wants [a PC] that runs not only types of applications that he knows he will want to use, but also those types in which he might develop an interest later. Also the consumer knows that if he chooses [a PC] with enough demand to support multiple applications in each product category, he will be less likely to find himself straitened later by having to use an application whose features disappoint him . . . [T]he average user knows that,

discourage the innovation of competing, incompatible products. The nature of the market insists on compatibility. Yet, the Patent Act bars any development of competing compatible products through the use of functional claiming. Functional claiming allows "[a]n element in a claim for a combination [to] be expressed as a means or step for performing a specified function without the recital of structure, material, or acts."¹⁴³ This grants monopoly rights to the patent holder for the element that is functionally claimed, barring any practical competition. "[W]hile a niche . . . system might turn a profit, the chicken-and-egg problem [*applications barrier to entry*] would make it prohibitively expensive for a new [system] to attract enough developers and consumers to become a viable alternative to a dominant incumbent in less than a few years."¹⁴⁴ Thus, a *positive feedback loop* is created that will continue to discourage entry of competitors and therefore discourage innovation.¹⁴⁵ A dominant PC's "positive feedback loop is for would be competitors a vicious cycle,"¹⁴⁶ a cycle that is reinforced through the functional claim aspect of patents.

Prior to 1986 the applications barrier to entry created a problem for consumers and for the hardware and software developers. Many consumers made investments in computer hardware incorporating different ROM BIOS codes for which there may have been only limited software compatibility.¹⁴⁷ Some buyers were unaware of the

generally speaking, applications improve through successive versions. He thus wants [a computer] system for which successive generations of his favorite applications will be released—promptly at that.

Id. at 19-20.

¹⁴³ 35 U.S.C. 112 (1994).

¹⁴⁴ *Microsoft Corp.*, 84 F. Supp. 2d at 25-26.

¹⁴⁵ *Id.* at 20. In addition,

[c]onsumer demand for [a standard PC] enjoys positive network effects. A positive network effect is a phenomenon by which the attractiveness of a product increases with the number of people using it. The fact that there is a multitude of people using [a particular type of PC] makes the product more attractive to consumers. The large installed base attracts corporate customers who want to use [a PC] that new employees are already likely to know how to use, and it attracts academic consumers who want to use software that will allow them to share files easily with colleagues at other institutions [A] large body of applications thus reinforces demand [for a particular type of PC], augmenting [that particular type of PC's] dominant position and thereby perpetuating [the independent software vendors'] incentives to write applications principally for [that PC]. This self-reinforcing cycle is often referred to as a 'positive feedback loop'.

Id. at 20.

¹⁴⁶ *See id.*

¹⁴⁷ Bullard, *supra* note 14, at 236.

possible incompatibility because many of the computer manufacturers were using similar, but not identical, microprocessors and operating systems. This incompatibility is similar to latent defects in real property which have the effect of "greatly impairing both the value of the property and its potential."¹⁴⁸ Because it would have been cost prohibitive for each consumer to determine the software compatibility of each computer the consumer considered purchasing, applying a strict rule of *caveat emptor* to the early PC sales would have been inappropriate absent the availability of extensive compatibility reporting. This uncertainty did not help promote the personal computer industry.

As the IBM PC became the *de facto* industry-wide standard, its positive feedback loop provided the incentive for most software companies to concentrate on developing software for that platform.¹⁴⁹ Because of the applications barrier to entry, hardware competitors would have effectively been barred from entering the market but for the availability of the functionally equivalent Phoenix BIOS software. The U.S. Department of Commerce reported that "a major technological development in 1986 hastened the trend toward commodity-like microcomputers. Several U.S. firms licensed low-cost compatible clones of the basic input/output software originally developed for the IBM PC. These independently developed programs perform[ed] substantially the same operating system functions yet d[id] not violate the developer's proprietary rights."¹⁵⁰

The benefits of this standardization could have been predicted. Standards "are essential components of our nation's technology infrastructure—vital to industry and commerce."¹⁵¹ Here, BIOS standardization eliminated the latent defect of the clones and "[s]oftware compatibility gained in importance as a factor underlying both software and hardware decision purchases, especially those made by large corporate buyers."¹⁵²

Some commentators believe that software differs from industrial products because of the relative investments that are required for commercialization and because the relative investment required for software commercialization makes patent protection inappropriate.¹⁵³

¹⁴⁸ See, e.g., *Stambovsky v. Ackley*, 572 N.Y.S.2d 672, 674 (App. Div. 1991).

¹⁴⁹ U.S. DEP'T OF COMMERCE, U.S. INDUSTRIAL OUTLOOK: (1987) - COMPUTERS AND SOFTWARE, at 28-7.

¹⁵⁰ *Id.* at 28-8.

¹⁵¹ Kammer, *supra* note 140.

¹⁵² U.S. DEP'T OF COMMERCE, *supra* note 149, at 28-8.

¹⁵³ Daughtrey, *supra* note 10, at 178.

In any event, the virtual nature of software, combined with the patent system's functional claiming provision is an effective bar to the development of any functionally equivalent products.¹⁵⁴

VI. DISCUSSION

Like other technical standards, the *de facto* BIOS standards helped to "diffuse new technologies," while "lower[ing] barriers to market entry."¹⁵⁵ This result is consistent with the intent of U.S. intellectual property laws.

Article I, Sec. 8 of the Constitution provides that: 'The Congress shall have power . . . to Promote the Progress of Science and useful Arts, by securing for limited Times to authors and Inventors the exclusive Right to their respective Writings and Discoveries.' The monopoly privileges that Congress may authorize are neither unlimited nor primarily designed to provide a special private benefit. Rather, the limited grant is a means by which an important public purpose may be achieved. It is intended to motivate the creative activity of authors and inventors by the provision of a special reward, and to allow the public access to the products of their genius after the limited period of exclusive control has expired. 'The copyright law, like the patent statute, makes reward to the owner a secondary consideration. 'The sole interest of the United States and the *primary objective in conferring the monopoly lie in the general benefits derived by the public* from the labors of authors.' . . . As the text of the Constitution makes plain, it is Congress that has been assigned the task of defining the scope of the limited monopoly that should be granted to authors or to inventors in order to give the public appropriate access to their work product.' In enacting a copyright [or patent] law Congress must consider . . . two questions: First, how much will the legislation stimulate the producer and so benefit the public, and, second, how much will the monopoly granted be detrimental to the public? The granting of such exclusive rights, under the proper terms and conditions, confers a benefit upon the public that outweighs the evils of the temporary monopoly.¹⁵⁶

¹⁵⁴ See THE NEW SHORTER OXFORD ENGLISH DICTIONARY, *supra* note 11.

¹⁵⁵ Kammer, *supra* note 140. See also U.S. DEP'T OF COMMERCE, U.S. INDUSTRIAL OUTLOOK (1989) - COMPUTER SOFTWARE, at 26-10 ("The inevitable spread of computer industry standards worldwide [is] a 'double-edged sword' for U.S. manufacturers. Standards can expand markets, lower prices, and allow users to have greater freedom of choice among vendors. On the other hand, standards can benefit some foreign suppliers by helping them to use their strength in high-volume, low-cost production to capture markets from their U.S. competitors.").

¹⁵⁶ Sony Corp. of Am. V. Universal City Studios, Inc., 464 U.S. 417, 428-29 (1984) (emphasis added) (citations omitted). Note that "[t]he Supreme Court treats patent and copyright the same

"The productive effort thereby fostered will have a positive effect on society through the introduction of new products and processes of manufacture into the economy, and the emanations by way of increased employment and better lives for our citizens."¹⁵⁷ The key that establishes this balance in intellectual property protection of software lies in the Constitution's intent "[t]o Promote the Progress of Science and useful Arts."¹⁵⁸

Any policy analysis regarding patent coverage of software must recognize "that the patent law encourages competitors to design or invent around existing patents."¹⁵⁹ First, "[d]esigning or inventing around patents to make new inventions is encouraged."¹⁶⁰ This "incentive to design around patents is a positive result of the patent system."¹⁶¹ This provides an opportunity for competitors to develop realistic products giving consumers more choices at lower prices.

[K]eeping track of a competitor's products *and designing new and possibly better or cheaper functional equivalents is the stuff of which competition is made and is supposed to benefit the consumer.* One of the benefits of a patent system is its so-called 'negative incentive' to 'design around' a competitor's products, even when they are patented, thus bringing a steady flow of innovations to the marketplace. It should not be discouraged . . .
 „¹⁶²

These features of the patent system help to promote commerce and the productivity of society as a whole. Second, the availability of a variety of products having a known functionality benefits the consumer because it allows for greater efficiency by eliminating new product training and by increasing the interchangeability of files.

Because of the interoperability requirement, the intent of the patent law is frustrated when computer software patents consist only of 35 U.S.C. § 112, ¶ 6 means-plus-function or step-plus-function claims. Any attempt to create a similar, but functionally different program would be futile because of the applications barrier to entry. Similar but functionally different programs would not enjoy many of

when looking at the purposes behind the Constitutional provision and the laws thereby enacted." Cass County Music Co. v. C.H.L.R., Inc., 88 F.3d 635, 641 (8th Cir. 1996).

¹⁵⁷ Int'l Techs. Consultants v. Pilkington PLC, 137 F.3d 1382, 1392 (9th Cir. 1998).

¹⁵⁸ U.S. CONST. art. I, § 8.

¹⁵⁹ WMS Gaming Inc. v. Int'l Game Tech., 184 F.3d 1339, 1355 (Fed. Cir. 1999).

¹⁶⁰ Westvaco Corp. v. Int'l Paper Co., 991 F.2d 735, 745 (Fed. Cir. 1993).

¹⁶¹ Yarway Corp. v. Eur-Control USA, Inc., 775 F.2d 268, 277 (Fed. Cir. 1985).

¹⁶² State Indus., Inc. v. A.O. Smith Corp., 751 F.2d 1226, 1235-1236 (Fed. Cir. 1985) (emphasis added). See also *Westvaco*, 991 F.2d at 745.

the features that compatible software would offer customers. For example, compatible software would allow customers to avoid the waste associated with learning to use new software packages, and it would provide customers with more software options to purchase for their systems.¹⁶³ Compatible systems may also facilitate the creation of networks and file sharing.¹⁶⁴

The U.S. Patent and Trademark Office attempted to codify judicial interpretation of 35 U.S.C. § 112, ¶ 6 software claims in published guidelines.¹⁶⁵ These guidelines classify a useful machine or manufacture according to its physical structure as a statutory product. Thus, claims for a useful machine or manufacture are evaluated independent of any accompanying process claims.¹⁶⁶ Patentability of statutory machine or manufacture embodiments of a process are evaluated on the basis of the product claims.¹⁶⁷

Under the guidelines for the Patent Act, process claims that involve a physical transformation outside of the computer are statutory subject matter under 35 U.S.C. § 101. When no physical transformation outside the computer occurs, a process comprises statutory subject matter only if there is a "claim to a practical application in the technological arts."¹⁶⁸ Examples of practical applications of computer software process claims include (1) a computerized method for controlling memory transfer between the cache and a hard disk; (2) a method of controlling multi-tasking parallel processors; (3) a method to store the executable code of a word processor in computer memory and (4) a digital noise filtering process.¹⁶⁹

If the 'acts' of a claimed process manipulate only numbers, abstract concepts or ideas, or signals representing any of the foregoing, the acts are not being applied to appropriate subject matter. Thus, a process consisting solely of mathematical operations, i.e., converting one set of numbers into another set of numbers, does not manipulate appropriate subject matter and thus cannot constitute a statutory process.¹⁷⁰

¹⁶³ Daughtrey, *supra* note 10, at 173-176.

¹⁶⁴ *See id.*

¹⁶⁵ Examination Guidelines for Computer Related Inventions, 61 Fed. Reg. 7478 (Feb. 28, 1996).

¹⁶⁶ *Id.* at 7482.

¹⁶⁷ *See id.*

¹⁶⁸ *Id.* at 7484.

¹⁶⁹ *Id.* at 7484.

¹⁷⁰ *See id.*

The Federal Circuit expanded the application of 35 U.S.C. § 112, ¶ 6 claims significantly in *State Street Bank & Trust Co. v. Signature Financial Group*, when it held that calculation of discrete dollar amounts constituted a practical application of a mathematical algorithm, formula or calculation.¹⁷¹ This expanding scope of patent coverage is inconsistent with the Constitutional objectives. IBM probably could have obtained patent coverage for their BIOS under the pre-*State Street* case law. Yet, *State Street* relaxed the requirements for software patents even further. This would have hindered the explosion in information technology that we are now experiencing.

IBM's production was inadequate to match demand for PCs and IBM had refused to grant licenses for their BIOS to their competitors. Had IBM patented their BIOS, competing computers would have been required to use functionally different BIOS programs, which would have lead to limited inter-compatibility of programs and files. This undoubtedly would have discouraged some consumers and software application vendors from making investments in a computer platform. Those who did make the investments would have done so in an inefficient manner because of the uncertain compatibility of shared files, or because of the delay that would have occurred as other *de facto* standards began to evolve.

Even though IBM did not have a patent on its ROM BIOS, IBM did remain competitive in the PC markets, and was not overtaken by technology freeloaders who profited from IBM's innovation without investment despite being protected not by patent, but by copyright. In 1996 IBM was still one of the three leading suppliers of personal computers.¹⁷² IBM apparently has not been discouraged from continuing its computer engineering efforts; in 1996 IBM invested nearly four billion dollars in research and development, making them the leading developer of advanced technologies.¹⁷³ In fact, consumers benefited from the competition enabled by the availability of the Phoenix BIOS. This year, the U.S. Department of Commerce is projecting that "[s]evere price competition among U.S. and foreign suppliers should make low-end systems more affordable and increase purchases of PCs in markets where penetration rates are currently

¹⁷¹ *State St. Bank & Trust Co. v. Signature Fin. Group*, 149 F.3d 1368, 1374 (Fed.Cir. 1998).

¹⁷² INT'L TRADE ADMIN., U.S. DEP'T OF COMMERCE, U.S. INDUS. & TRADE OUTLOOK '98 (1998), at 27-10. The other two leaders were Compaq and the NEC/Packard Bell/Zenith joint venture.

¹⁷³ INT'L TRADE ADMIN., U.S. DEP'T OF COMMERCE, U.S. INDUS. & TRADE OUTLOOK '99 (1999), at 27-4.

much lower than they are in the United States and where disposable incomes are rising.”¹⁷⁴ Not only has the cost decreased as one would expect from improved economies of scale and recoupment of investment in facilities, the capabilities of available equipment have also increased.

When the first IBM PC was unveiled in 1981, priced at \$4,500, it had only limited processing power, main memory, one operating system, and no hard disk or cache memory. In 1996, 15 years later, IBM's PC300 series sold for \$1,400 less and featured a microprocessor with 40 times the processing power, a huge main memory that allowed the central processor to handle many complex tasks, and several choices of very sophisticated, more user-friendly operating systems. The system also had more than half a megabyte (one million bytes) of cache memory, disk storage comparable to that sold with mainframes only a decade ago, and a large-screen, high-resolution monitor. This price/performance should continue unabated through the turn of the century, with the most significant price declines expected in low-end computer systems and peripherals.¹⁷⁵

This is exactly the result the drafters of our Constitution intended. Because the courts can only decide cases based on the statutes as written, Congress should make appropriate changes to 35 U.S.C. § 101 and 35 U.S.C. § 112 to prohibit software patents. Computer programs should continue to be protected under the copyright laws as the two federal advisory commissions have recommended. This would make U.S. protection of software consistent with international laws,¹⁷⁶ an important consideration recognized by President Johnson in 1965 when he convened the President's Commission on the Patent System to “ensure that the patent system . . . [was] effective in serving the public interest in view of the complex and rapidly changing technology of [the] time.”¹⁷⁷

The copyright laws are free of the patent system's difficulties in determining prior art for software inventions as anticipated by the President's Commission on the Patent System.¹⁷⁸ One recent example of these difficulties is illustrated by a windowing fix to the Y2K

¹⁷⁴ *Id.* at 27-7.

¹⁷⁵ INT'L TRADE ADMIN., U.S. DEP'T OF COMMERCE, U.S. INDUS. & TRADE OUTLOOK '98 (1998), at 27-2.

¹⁷⁶ Council Directive 91/250, 1991.

¹⁷⁷ Exec. Order No. 11215, (Apr. 10, 1965), 1965 WL 7882.

¹⁷⁸ PRESIDENT'S COMMISSION ON THE PATENT SYSTEM, TO PROMOTE THE PROGRESS OF . . . USEFUL ARTS: IN AN AGE OF EXPLODING TECHNOLOGY at 12. (1966).

millennium bug patented by Bruce Dickens.¹⁷⁹ Dickens' approach assigns a time window to a two-digit date to determine the actual date, "by assigning numbers a chosen point, say '50,' to the 20th century, and numbers below that point to the 21st. So '51' would be read as '1951,' whereas '49' would be read as '2049.'"¹⁸⁰ However, some reports indicate that similar "windowing was used as early as the late 1960s."¹⁸¹ If such prior art actually exists, the patent was issued presumably due to the examiner's difficulty in locating the relevant prior art. Because there was considerable concern regarding the validity of the patent related to this possible prior art,¹⁸² the Commissioner of Patents and Trademarks issued a rare order for reexamination.¹⁸³ "If every Fortune 500 corporation opted to [agree to the licensing] fee, [the assignee] would wind up with between \$165 million and \$16.5 billion—depending on when the companies pay—for just the first year of the patent's 17 [sic] year life."¹⁸⁴

At least one law professor believes that "blindingly obvious" Internet or software patents such as the Dickens' patent "will have a chilling effect on electronic commerce."¹⁸⁵ This is inconsistent with the constitutional intent for the patent system "[t]o promote the Progress of Science and useful Arts."¹⁸⁶ This is also inconsistent with the CONTU recommendation that the intellectual property system should not grant anyone more economic power than is necessary to achieve the incentive to create.

IBM recognized the value of software copyright protection; from 1983 to 1985 IBM registered copyrights for its personal computer BIOS and its updates at least seven times.¹⁸⁷ Signature Financial, the

¹⁷⁹ U.S. Patent No. 5,806,063 (issued Sept. 8, 1998).

¹⁸⁰ Richenya A. Shepherd, *Y2K Fixers are Outraged by Patent Payment Demand; Programmer Seeks Royalties for 'Windowing,' but his Rivals Say it Preceded Patent*, NAT'L L.J., at B9 (Dec. 13, 1999).

¹⁸¹ Ghandrasekaran, *supra* note 128, at E1. See also Shepherd, *supra* note 180, at B9.

¹⁸² Ghandrasekaran, *supra* note 128, at E1; Shepherd, *supra* note 180, at B9.

¹⁸³ PRESS RELEASE #99-51, PATENT AND TRADEMARK OFFICE U.S. DEP'T OF COMMERCE, PATENT AND TRADEMARK OFFICE ORDERS REEXAMINATION OF Y2K FIX PATENT (Dec. 21, 1999).

¹⁸⁴ Ghandrasekaran, *supra* note 128, at E1.

¹⁸⁵ See *id.*

¹⁸⁶ U.S. CONST. art. I, § 8, cl. 8.

¹⁸⁷ U.S. Copyright Registration No. TX1178238 (issued 1983); U.S. Copyright Registration No. TX1363979 (issued 1984); U.S. Copyright Registration No. TX1363981 (issued 1984); U.S. Copyright Registration No. TX1481758 (issued 1985); U.S. Copyright Registration No. TX142257 (issued 1984); U.S. Copyright Registration No. TX1434859 (issued 1984); U.S. Copyright Registration No. TX1730028 (issued 1985). Note that Phoenix Technologies also registered the copyrights on their compatible BIOS. See, e.g., U.S. Copyright Registration No.

assignee of the patent litigated in *State Street Bank*, also recognized the importance of copyright registration of its software. In 1990 and 1991 it registered the copyright of two computer programs to cover software modules that were the subject of the Boes patent.¹⁸⁸

One might expect the owner of a proprietary software innovation, like Signature Financial, to seek all forms of intellectual property protection that are available. Surprisingly, this was not the case for the examples cited above.¹⁸⁹ Federal-Mogul Corporation, employer of Diehr and Lutton, did not register a copyright for the software at issue in *Diehr*.¹⁹⁰ Federal-Mogul attorneys considered the Diehr and Lutton innovation to relate more to the process rather than to the software and did not believe such registration was necessary.¹⁹¹ Tektronix, Inc., the employer of Alappat, Ayerill and Larson, also did not register the software at issue in *Alappat*.¹⁹² Tektronix did not consider such registration to be necessary because their code was unique to their already patented hardware.¹⁹³ The copyright for the programs at issue in *Lowry*¹⁹⁴ and *Excel Communications*¹⁹⁵ also were not registered, presumably for similar reasons. Note that the copyright statutes protect all of these software innovations even though registrations have not previously been filed.¹⁹⁶

VII. CONCLUSION

The exclusion of software patent claims from statutory subject matter does not deny intellectual property protection to software creators. Significant intellectual property protection is already available for original software when a copyright for that software is

TX1835932 (issued 1986).

¹⁸⁸ Telephone interview with Molly S. Mugler, Senior Legal Counsel, Signature Financial Group, Inc. (Aug. 11, 2000). U.S. Copyright Registration Nos. TX422906 (1990), TX3207622 (1991). These registrations occurred contemporaneously with their March 11, 1991 filing of U.S. Patent No. 5,193,056 (issued Mar. 9, 1993).

¹⁸⁹ See *supra* text accompanying notes 104-127.

¹⁹⁰ Telephone interview with Melville Owen, Attorney, Owen, Wickersham & Erickson, counsel for U.S. Patent at issue, U.S. Patent No. 4,344,142 (issued June 8, 1982) (July 25, 2000).

¹⁹¹ See *id.*

¹⁹² Telephone interview with Francis I. Gray, Tektronix patent counsel for U.S. Patent at issue 5,440,676 (issued Aug. 8, 1995) (July 25, 2000).

¹⁹³ See *id.*

¹⁹⁴ Telephone interview with Kenneth Kozik, Fish & Richardson, counsel for U.S. Patent at issue 5,664,177 (Sept. 2, 1997) (July 26, 2000).

¹⁹⁵ Telephone interview with Ronald D. Slusky, patent counsel for U.S. Patent at issue 5,333,184 (issued July 26, 1994) (July 27, 2000).

¹⁹⁶ See, e.g., 17 U.S.C. § 408 (1994).

appropriately registered. First, the copyright code prohibits the production of derivative works, so the expensive clean room techniques described above are necessary to legitimately produce competing products. Second, if the copyright is registered prior to infringement, or within three months of publication, statutory damages and attorney fees can be awarded in the event of a finding of infringement.¹⁹⁷ A software creator can also bring a cause of action for infringement of unregistered software if a copyright is registered prior to the filing of the action. In this case, the copyright owner may recover actual damages, the profits of the infringer and obtain an injunction.¹⁹⁸

The Patent Act should be amended to bar software patents. Functional claiming, an integral part of our patent system since 1952, predates the concept of virtual inventions like software.¹⁹⁹ However, functional claiming, combined with the virtual nature of software, is an absolute bar to the development of any functionally equivalent software product. In addition, "the PTO's minimal disclosure requirements in combination with the availability of means-plus-function language for software algorithm claims has resulted in the issuance of patents that add little if nothing to the public domain, but which may be interpreted as affording a broad right of exclusion to the patent holder."²⁰⁰ Our intellectual property laws "involve a difficult balance between the interests of authors and inventors in the control and exploitation of their writings and discoveries on the one hand, and society's competing interest in the free flow of ideas, information, and commerce on the other hand, [which is why] our patent and copyright statutes have been amended repeatedly."²⁰¹ It is now time for such an amendment.

¹⁹⁷ 17 USC § 412 (1994).

¹⁹⁸ John W. Hazard, Jr., COPYRIGHT L. BUS. & PRAC. § 5.06 (1998). Note that copyright protection is not dependent on prior registration. Cf. 17 U.S.C. § 408 (Rev. ed. 1994). However, registration is required before any infringement action can be instituted. Cf. 17 U.S.C. § 411 (1994 & Supp. IV 1998).

¹⁹⁹ See, e.g., *Valmont Indus., Inc. v. Reinke Mfg. Co., Inc.*, 983 F.2d 1039, 1041-1044 (Fed. Cir. 1993).

²⁰⁰ Shawn McDonald, *Patenting Floppy Disks, or How the Federal Circuit's Acquiescence Has Filled the Void Left By Legislative Inaction*, 3 VA. J.L. & TECH. 9, para. 6, n.8 (1998).

²⁰¹ *Sony Corp. of Am. V. Universal City Studios, Inc.*, 464 U.S. 417, 429 (1984).

